**PATENT**

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

| | | |
|---|---|---|
| In re application of:<br>Prashant Modi | ) | Examiner: Wanda Z. Russell |
| | ) | Art Unit: 2616 |
| Serial No.: 10/806,628 | ) | |
| Filed: 03/23/2004 | ) | Confirmation No: 2628 |
| For: **Pre-Configured Topology With Connection Management** | ) | |
| Date of Final Office Action:<br>July 3, 2008 | ) | Attorney Docket No.:<br>200312794-1 |
| Notice of Appeal Filed:<br>August 11, 2008 | ) | |

September 29, 2008

**APPEAL BRIEF**

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This Appeal Brief is timely provided to support the Notice of Appeal filed August 11, 2008.

**1. Real Party In Interest:**

The real party in interest is Hewlett-Packard Development Company, LP, a limited partnership established under the laws of the State of Texas and having a principal place of business at 20555 S.H. 249 Houston, TX 77070, U.S.A. (hereinafter "HPDC"). HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware Corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, LLC.

## 2. Related Appeals and Interferences

There are no other prior and/or pending appeals, interferences, or judicial proceedings that are related to, directly affect, or that will be directly affected by or have a bearing on the Board's decision.

## 3. Status of Claims

Claims 1-39 are pending in the application.

Claims 1-39 stand rejected.

No claims were canceled.

No claims were allowed.

No claims were withdrawn

The rejections of claims 1-39 are appealed.

## 4. Status of Amendments

No Amendments were filed subsequent to the Final Office Action.

## 5. Summary of Claimed Subject Matter

Independent Claim 1

Claim 1 concerns a system. (p9, [31], l1, Fig. 1, element 100). The system includes an interface logic. (p9, [31], l2-3, Fig. 1, element 110) The interface logic is configured to pre-configure a topology of nodes to communicate via a preferred networking protocol. (p9, [31], l2-3, Fig. 1, element 110)

The system also includes a mapping logic operably connected to the interface logic. (p10, [36], l1-8, Fig. 1, element 122) The mapping logic is configured to produce a mapping between a resource located on a first node and a port located on the first node. (p10, [36], l2-4, Fig. 1, element 122) The mapping logic is to selectively provide to a second node a mapping data that describes the mapping. (p10, [36], l7-9, Fig. 1, element 122). The mapping logic is to selectively establish a connection that facilitates the second node accessing the resource through the port using the preferred networking protocol. (p11, [37], l1-3, Fig. 1, element 122)

The system also includes a connection management logic operably connected to the mapping logic and the interface logic. (p11, [38], l1-2, Fig. 1, element 124). The connection management logic is to control whether the mapping logic will provide the mapping data and establish the connection. (p11, [38], l3-5, Fig. 1, element 124)

Independent Claim 17

Claim 17 concerns a computer. (p18, [66], l1, Fig. 8, element 800) The computer is configured with a pre-configured topology connection management system. (p9, [31], l1-3, Fig. 1, element 100), (p18, [66], l1-7, Fig. 8, element 830). The system includes an interface logic configured to pre-configure a

topology of nodes to communicate via a preferred networking protocol or a fallback networking protocol. (p9, [31], l2-3, Fig. 1, element 110) To pre-configure the topology of nodes, the interface logic acquires a node identifier that facilitates recording whether a node is a member of a pre-configured topology. (p9, [32], l1-7). To pre-configure the topology of nodes, the interface logic also acquires a topology configuration choice data concerning how the pre-configured topology is to be configured. (p9, [33], l1-3) The interface logic pre-configures the topology based, at least in part, on the node identifier and the topology configuration choice data, and provides a topology data concerning the topology to a member of the topology. (p9, [32], l8-11)

The system also includes a mapping logic operably connected to the interface logic. (p10, [35], l1-3, Fig. 1, element 122). The mapping logic is to produce a mapping between a resource located on a first node and a port located on the first node. (p10, [36], l2-4, Fig. 1, element 122). The mapping logic is to selectively provide to a second node a mapping data that describes the mapping between the resource and the port. (p10, [36], l7-9, Fig. 1, element 122) The mapping logic is to selectively establish a connection between the first node and the second node, where the connection facilitates the second node accessing the resource through the port using the preferred networking protocol. (p11, [38], l3-5, Fig. 1, element 124)

The system includes a connection management logic operably connected to the mapping logic and the interface logic. (p11, [38], l1-2, Fig. 1, element 124) The connection management logic is to control whether the mapping logic will provide the mapping data to the second node, and whether the mapping logic will establish the connection. (p11, [38], l3-5, Fig. 1, element 124) The connection management logic exerts its control based, at least in part, on the topology data and a node identifier received from the second node.

Independent Claim 18

Claim 18 concerns a method. (p14, [47], l1, Fig. 3, element 300). The method includes acquiring a set of node identifiers associated with nodes to be considered for inclusion in a pre-configured topology of nodes that can communicate within the topology using a preferred computer networking protocol. (p14, [47], l2-11, Fig. 3, element 310) The method also includes establishing the pre-configured topology of nodes. (p14, [48], l1-9, Fig. 3, element 320) The method also includes making available a membership data concerning the pre-configured topology of nodes. (p14, [49], l1-4, Fig. 3, element 330)

Independent Claim 22

Claim 22 concerns a computer-readable medium storing processor executable instructions operable to perform a method. (p14, [50], l1-6). The method includes acquiring a set of node identifiers associated with nodes to be considered for inclusion in a pre-configured topology of nodes that can communicate within the topology using a preferred computer networking protocol or a fallback computer networking protocol. (p14, [47], l2-11, Fig. 3, element 310). The method also includes establishing the pre-configured topology of nodes, where establishing the pre-configured topology of nodes includes determining node membership in the pre-configured topology, establishing a preferred computer networking protocol to be employed by members of the topology, establishing a preferred path to be employed for communications between members of the topology, establishing a fallback computer networking protocol to be employed by members of the topology, establishing a fallback path to be employed for communications between members of the topology, and recording the topology membership, preferred computer networking protocol, preferred path, fallback computer networking protocol, and fallback path in the membership data. (p14, [48], l1-9, Fig. 3,

element 320) The method also includes making available a membership data concerning the pre-configured topology of nodes. (p14, [49], l1-4, Fig. 3, element 330)

Independent Claim 23

Claim 23 concerns a method. (p15, [51], l1, Fig. 4, element 400) The method includes acquiring a set of node identifiers associated with nodes to be considered for inclusion in a pre-configured topology of nodes that can communicate within the topology using a preferred computer networking protocol. (p15, [51], l2-8, Fig. 4, element 410) The method also includes establishing the pre-configured topology of nodes. (p15, [52], l1-6, Fig. 4, element 420) The method also includes distributing a membership data concerning the pre-configured topology of nodes to nodes that are in the pre-configured topology of nodes. (p15, [53], l1-6, Fig. 4, element 430) The method also includes selectively adding or deleting a node from the pre-configured topology of nodes. (p15, [54], l1-4, Fig. 4, element 450) The method also includes, in response to selectively adding or deleting the node, redistributing the membership data. (p15, [54], l2-5, Fig. 4, element 460). The method also includes selectively managing a computer networking resource. (p16, [55], l5-9, Fig. 4, element 480). The method also includes, in response to selectively managing the computer networking resource, redistributing the membership data. (p16, [55], l5-7, Fig. 4, element 490)

Independent Claim 27

Claim 27 concerns a method. (p16, [57], l1, Fig. 5, element 500) The method includes receiving, in a first node, from a second node, via an open computer networking protocol, a request to establish a connection between the first node and the second node via the open computer networking protocol,

where the connection facilitates the second node accessing a resource associated with the first node. (p16, [57], l4-8, Fig. 5, element 510) The method also includes determining whether the second node is a member of a pre-configured topology that includes the first node by examining a node identifier associated with the second node. (p16, [58], l1-5, Fig. 5, element 520) The method also includes selectively not establishing the connection between the first node and the second node via the open computer networking protocol if it is determined that the second node is not a member of the pre-configured topology that includes the first node. (p16, [58], l1-7)

Independent Claim 30

Claim 30 concerns a computer-readable medium storing processor executable instructions operable to perform a method. (p14, [50], l1-6). The method includes, in a session layer logic in a first node, receiving from a second node, via an open computer networking protocol that includes a Transmission Control Protocol (TCP) transport layer and an Internet Protocol (IP) network layer, a request to establish a connection between the first node and the second node via the open computer networking protocol, where the connection facilitates the second node accessing a resource associated with the first node. (p16, [57], l4-8, Fig. 5, element 510). The method also includes determining whether the second node is a member of a pre-configured topology that includes the first node. (p16, [58], l1-5, Fig. 5, element 520) The method also includes selectively not establishing the connection between the first node and the second node via the open computer networking protocol if it is determined that the second node is not a member of the pre-configured topology that includes the first node. (p16, [58], l1-7)

Independent Claim 31

Claim 31 concerns a method. (p17, [59], l1, Fig. 6, element 600) The method includes, in a first node, receiving from a second node a mapping request for a mapping data that describes a relationship between a resource on the first node and a port on the first node. (p17, [59], l3-13, Fig. 6, element 610). The method also includes selectively providing the mapping data to the second node based on determining that the second node is a member of a pre-configured topology that includes the first node by examining a node identifier associated with the second node. (p17, [60], l1-5, Fig. 6, element 620-630) The method also includes receiving from the second node a connection request to establish a connection between the first node and the second node via a first networking protocol, where the connection facilitates accessing the resource. (p17, [61], l1-5, Fig. 7, element 710) The method also includes selectively establishing the connection based on determining that the second node is a member of a pre-configured topology that includes the first node by examining a node identifier associated with the second node. (p17, [62], l1-6, Fig. 7, element 720-730) The method also includes, via a second networking protocol, receiving from the second node a fallback connection request to establish a fallback connection between the first node and the second node. (p18, [63], l1-7, Fig. 7, element 740). The fallback connection request requests that the fallback connection be established via the second networking protocol, where the fallback connection granted in response to the third request will not provide access to the resource via the first networking protocol. (p18, [63], l1-7, Fig. 7, element 750-760)

Independent Claim 36

Claim 36 concerns a computer-readable medium storing processor executable instructions operable to perform a method. (p14, [50], l1-6). The method includes in a first node, receiving from a second node a mapping

request for a mapping data that describes a relationship between a resource on the first node and a port on the first node. (p17, [59], l3-13, Fig. 6, element 610) The method also includes selectively providing the mapping data to the second node based on determining, by examining a node identifier associated with the second node, that the second node is a member of a pre-configured topology that includes the first node. (p17, [60], l1-5, Fig. 6, element 620-630) The method also includes receiving from the second node a connection request to establish a connection between the first node and the second node via a first networking protocol, where the connection facilitates accessing the resource. (p17, [61], l1-5, Fig. 7, element 710) The method also includes selectively establishing the connection based on determining that the second node is a member of a pre-configured topology that includes the first node by examining a node identifier associated with the second node. (p17, [62], l1-6, Fig. 7, element 720-730) The method also includes, via a second networking protocol, receiving from the second node a fallback connection request to establish a fallback connection between the first node and the second node. (p18, [63], l1-7, Fig. 7, element 740). The fallback connection request requests that the fallback connection be established via the second networking protocol, where the fallback connection granted in response to the third request will not provide access to the resource via the first networking protocol. (p18, [63], l1-7, Fig. 7, element 750-760)

Independent Claim 37

Claim 37 concerns a system. (p18, [66], l1-7, Fig. 8, element 800, 830) Claim 37 recites means for determining whether a client node is a member of a pre-configured topology to which a server node belongs. One structure that corresponds to the claimed function is port mapping logic 830. (p18, [66], l2-9, Fig. 8, element 830) This claim also recites means for rejecting a request that will lead to the undesired consumption of a server resource if the requesting

client node is not a member of the pre-configured topology to which the server node belongs. One structure that corresponds to the claimed function is port mapping logic 830. (p18, [66], l2-9, Fig. 8, element 830) This claim also recites means for establishing a connection between the client node and the server node using a networking protocol preferred by members of the pre-configured topology. One structure that corresponds to the claimed function is port mapping logic 830. (p18, [66], l2-9, Fig. 8, element 830)

Independent Claim 38

Claim 38 concerns a set of application programming interfaces embodied on a computer-readable medium for execution by a computer component in conjunction with pre-configured topology connection management. (p20, [72], l1-4, Fig. 9, element 900) The set of application programming interfaces includes a first interface for communicating a group data configured to facilitate determining whether a client node is a member of a pre-configured topology to which a server node belongs. (p21, [73], l2-4, Fig. 9, element 940). The set of application programming interfaces also includes a second interface for communicating a resource management data that facilitates determining whether a client node will be granted a connection to a resource located on a topology member node. (p21, [73], l4-7, Fig. 9, element 950)

## 6. Grounds of Rejection to be Reviewed on Appeal

I. Whether claim 11 is unpatentable under 35 U.S.C. 103(a) as being obvious over Elzur et al. (Pub. No. US 2004/0093411 A1)(Elzur), in view of Delany et al. (U.S. Patent 6,658,454 B1)(Delany), and Wright et al. (Pub. No. US 2005/0154825).

II. Whether claims 1-3, 7-9, 12-18, 23-30, and 37-38 are unpatentable under 35 U.S.C. §102(e) as being anticipated by Elzur.

III. Whether claims 4-6, 10, 19-22, 31-36, and 39 are unpatentable under 35 U.S.C. 103(a) as being obvious over Elzur in view of Delany.

## 7. Argument

I. Whether claim 11 is unpatentable under 35 U.S.C. 103(a) as being obvious over Elzur, in view of Delany, and Wright.

This claim describes how a connection management logic can "block access to a first resource on the first node via the preferred networking protocol". The Final Office Action rejects this claim because Fair (incorrectly identified as Wright), includes, in line 11 of paragraph 42, the statement that "by supporting a plurality of block access protocols, the multiprotocol storage appliance provides ..." There is simply no nexus between the claim and the reference, except that some of the words in the claim appear in the reference.

The claim concerns allowing a node to communicate using a second (e.g., fallback) protocol after blocking that node from communicating using a first protocol. The reference concerns a disk device and describes its "block access protocols". A block is a unit of storage on a disk. A block access protocol is a method for reading/writing that unit of storage on a disk. Blocking access to a resource concerns preventing someone from using that resource. The claim and the reference are simply unrelated. This rejection demonstrates one of the pitfalls of keyword based rejections. A sentence that concerns "block access protocols" has been used to reject a claim that blocks access to a resource via a first protocol but grants access to the resource via a second protocol.

This keyword based rejection does not establish a prima facie case of obviousness for the claim. Thus the rejection should be reversed and the claim allowed.

II. Whether claims 1-3, 7-9, 12-18, 23-30, and 37-38 are unpatentable under 35 U.S.C. §102(e) as being anticipated by Elzur.

Independent Claim 37

Claim 37 stands "rejected by reference" to claim 27. However, claim 37 recites a system claim in means plus function language while claim 27 is a method claim. The claims do not share all the same elements and limitations. Thus, the elements and limitations of claim 37 have not been examined and a prima facie case for anticipation cannot possibly have been established. Therefore this rejection is improper and should be reversed, leaving claim 37 not anticipated and in condition for allowance.

Independent Claim 38

This claim was also "rejected by reference" to claims 27 and 37. Claim 27 is a method claim and claim 37 is a system claim. Claim 38 concerns neither a system nor a method, but rather an API for communicating group data and resource management data, neither of which are elements of claims 27 or 37. This "rejection by reference" does not provide Applicant with an adequate opportunity to reply and is, therefore, improper. Additionally, since the elements of an API have never been examined, it is impossible for a prima facie case for anticipation to have been made for this claim. Therefore this rejection should be reversed, leaving claim 38 not anticipated and in condition for allowance.

**The Claims Patentably Distinguish Over the References of Record**

**35 U.S.C. §102**

**Claims 1-3, 7-9, 12-18, 23-30 and 37-38** were rejected under 35 U.S.C. §102(e) as being anticipated by Elzur. For a 35 U.S.C. §102 reference to anticipate a claim, the reference must teach every element of the claim. Section 2131 of the MPEP recites:

> A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a

> single prior art reference. *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987).

Here, the reference cited in the Office Action, Elzur, does not teach every element of the rejected claims. Elzur does not teach the claimed logics. The Final Office Action states that "logic is a place to store program and/or data for execution." (Final Office Action, pg 17, lines 4-5). However, logic is defined on pg. 7, [24] as being "hardware, firmware, software, and/or combinations of each **to perform** a function(s) or an action(s), and/or to cause a function or action." (Emphasis added). Thus, the claimed logic takes or causes an action. The Final Office Action logic and the logic of Elzur can neither take nor cause an action. The logic of Elzur is a simple data structure.

Elzur is directed at a multi-tier data center that may handle multiple different traffic types over a single fabric. (Elzur, Page 1, Abstract). The data center produces systems with substantial power and space capabilities because the first tier may interface with secondary tiers to improve performance and reduce cost and complexity. (Elzur, Page 3, Paragraph [0034]). However, Elzur says nothing about pre-configuring a topology of nodes to communicate via a preferred network protocol.

Elzur describes a system and method for network interfacing. (Title) The summary describes the invention as providing a "data center" that includes several tiers. [0013] Elzur boldly asserts that it describes a device that can "handle all communication needs of a computer". [0033] These needs may be serviced by a TCP offload engine and an RDMA protocol that runs on top of TCP. [0033] The needs may also be met by a flow-through network interface card (NIC) that is optimized to minimize resources used to handle different traffic types and different interfaces. [0036] The network interface card may be multi-functional and support LAN traffic concurrently with TCP offload, iSCSI, and RDMA traffic. Clearly this RDMA and offload capable NIC is relevant to the claimed invention, but only as an

example of conventional systems that do not perform pre-configured topology membership based connection management.

The Office Action relies on Fig. 6 as describing a system that includes an interface logic (e.g., SCSI [0043]). However, this SCSI interface does not appear able to pre-configure a topology of nodes as claimed and described. While the SCSI interface can "operate directly on application data and run complete ... protocol stacks," [0006], it is not described as being able to pre-configure a topology of nodes as claimed and described.

The Office Action on page 2, second to last paragraph, asserts that Fig. 6 and [0041] teach pre-configuring a topology of nodes. This figure, this paragraph, and indeed the entire reference only teach conventional SCSI processing and are completely silent about pre-configuring as claimed and described. For at least this reason, independent claims 1, 17, 18, and 23 are not anticipated by Elzur and are in condition for allowance. Accordingly, dependent claims 2-3, 7-9, 12-16, and 24-26 are similarly not anticipated and are in condition for allowance.

The Office Action, in the Response to Arguments, on page 16, point 7, re-asserts that the applicant's term "pre-configuring a topology of nodes" is just a simple computer network taught in Figure 6 of Elzur. This is incorrect. The specification at [0016] introduces "topology of nodes." Nodes can be mapped to a topology. Nodes may be for example, physical ports and logical addresses associated with the physical ports, which are individually mapped to a topology. [0017] – [0021] go into great detail about "topology of nodes." [32] – [34] describe in great detail how interface logic 110 may pre-configure a topology of nodes. The Final Office Action asserts that "it is inherent that the interface (e.g., SCSI) is configured to work with the processor ... to configure ... associated devices." The Final Office Action has confused configuring a device with configuring a topology. Configuring a device makes that device ready to work. Configuring a device may include, for example, writing bits to configuration registers. Configuring a topology refers to establishing a group membership that can be used to control protocol decisions. The claim describes "configuring a topology", not configuring a device.

Paragraphs [32] – [34] describe in great detail how this includes acquiring identifiers and recording identifiers to later identify whether a node is a member of the topology. The nodes aren't configured by the interface logic, the interface logic configures a topology. Membership in the topology is then examined to determine whether certain RNIC operations will be allowed. The "inherent" configuring of computer devices or a SCSI interface relied on in the Final Office Action has nothing to do with configuring a topology in a way that facilitates membership based RNIC control as claimed and described. A careful examination of the details provided in [32]-[34] when compared to the circumspect "inherency" rejection reveal that a prima facie case for anticipation has simply not been made, leaving the claim allowable. Thus, Applicant respectfully requests reversal of the rejection and allowance of the claim.

Elzur teaches only conventional SCSI processing and is silent about pre-configuring a topology of nodes. On page 16, the Office Action recites:

> The term "preconfiguring a topology of nodes" applicant uses is just a computer network with associated devices such as disk. Fig. 6 of Elzur teaches exactly the "topology." Computers have been used for decades and *it is inherent* that the interface (e.g., SCSI or iSCSI) is configured to work with the processor of the computer to configure all or some of the associated devices. (Emphasis Added)

The "official notice" completely overlooks the recited elements and their complexity.

The Final Office Action on page 2, last paragraph and on pages 16 & 17, asserts that the command descriptor block (CDB) described in [0043] and [0008] is a mapping logic than can produce a mapping between a resource and a port on a first node. This is incorrect. The Final Office Action identifies the mapping logic as being the SCSI CDB ([43], l1-3), identifies the interface logic as being the SCSI HBA ([43], l1), and identifies the connection management logic as being the SCSI CNC ([24] l2, [40] l1-7). These data structures cannot possibly be operably connected as claimed. Even if the data structures could be operably connected, which they cannot, the amalgamation of data structures

still could only influence the operation of a program or processor. The impossible combination could not possibly pre-configure a topology of nodes (e.g., establish membership in a group) to facilitate later control of RNIC protocol operations like providing mapping data and/or establishing connections as claimed and described.

It is known in the art that the CDB is a data structure consisting of a one byte operation code followed by a few bytes of command-specific parameters. Therefore, a CDB is a data structure and/or interface upon which or through which actions may be performed. It is not a logic that does things. Therefore, it cannot possibly produce a mapping. It may store some data related to a mapping or allow the passing of some mapping data, but it cannot produce a mapping.

The Final Office Action also asserts that the CDB selectively provides mapping data to a second node. This is also incorrect. Since the CDB does not produce a mapping, it cannot possibly provide that mapping to a second node. Even if the CDB does store or permit the passage of a mapping that is provided to a second node, which it does not, it does not provide this mapping data selectively. To the extent the CDB provides any data, it does so non-selectively, according to conventional approaches that are unaware of pre-configured topology based connection management.

Additionally, on page 17, point 7, in the Response to Arguments the Final Office Action asserts that the recited mapping logic is taught in [0043]. The Office Action states "'During transmission, the host may get the SCSI CDB and the iSCSI context <u>for a connection</u>.' (emphasis added) It is clear that the SCSI CDB works with the processor of the computer to establish the connection that is 'mapping.'" This reflects an incorrect understanding of the claimed port mapping between a resource on a first node and a port on a first node. Claim 1 recites the mapping logic being configured to produce a mapping between a resource located on a first node and a port on the same node. The SCSI CDB does not "work with" the processor to establish a connection. The CDB is

simply a data structure that stores data describing a simple operation. The SCSI CDB is not mapping logic configured to produce a map between resources. Therefore, the CDB does not "work with" the processor. As stated above, the CDB is a data structure and\or interface. In the context of the stated quote, the host "gets" the SCSI CDB and iSCSI context "for the connection." This does not imply that SCSI CDB is the logic mapping the resources between a device on the node and a port on the node.

The Final Office Action also asserts that the CDB selectively establishes a connection that facilitates the second node accessing the resource through the port [0041] using the preferred network protocol. This is also incorrect. Fig. 6 and the CDB merely describe an iSCSI that may provide control and data transfer functions, but not selective connection establishment over a preferred protocol. The data transfer portion may build iSCSI protocol data units (PDUs) from SCSI CDBs, but this is not selectively establishing a connection over a preferred protocol as claimed and described. This is using a connection established in a standard way to move data structures (e.g., PDUs, CDBs) in a standard way. The data structures and/or interfaces are not logics. They are just data being moved around and/or the portals through which they move. For at least this additional reason independent claims 1 and 17 are not anticipated by Elzur and are, therefore, in condition for allowance. Accordingly, dependent claims 2-16 and 18 are also not anticipated and are in condition for allowance.

Additionally, on page 17-18, point 8, in the Response to Arguments, the Final Office Action asserts that the selective connection establishment over a preferred protocol is described in Elzur in [0033], lines 9-10, [0038], lines 8-9, and [0041], lines 7-8. In each of these cited passages the reference is merely stating that a device can communicate using standard network connections (TCP, iSCSI, RDM, multiple TCP ....). The cited sections of the reference do not teach a selective connection establishment over a "preferred protocol" as described and claimed. A reference that simply recites several different protocol names does not anticipate a claim that controls how and whether a

connection will be made at an RNIC based on group membership in a pre-configured topology of nodes. For at least this additional reason, independent claims 1 and 17 are not anticipated by Elzur and are in condition for allowance.

The Final Office Action asserts that [0024] teaches a connection management logic that controls where the mapping data will be provided and whether the connection will be established. This is incorrect. The Office Action relies on the converged network controller (CNC) illustrated in Fig. 7 and described in [0040]. Additionally, on page 17, point 9, in the Response to Arguments, the Office Action asserts that the selective connection establishment over a preferred protocol is described in Elzur in [0040] and Figure 7. The CNC may construct TCP segments, compute a CRC, insert a marker, and so on. However, the CNC does not control whether the mapping logic will provide mapping data and/or establish a connection. It simply does standard processing like that associated with prior art systems. Indeed, using the reasoning of the Final Office Action concerning a logic being "a place to store program and/or data for execution", it is physically impossible for the CNC to do what is asserted in the Final Office Action. The CNC would have to be able to control the mapping logic to provide data and establish a connection. There is no conceivable way the CNC could cause the purported mapping logic (e.g., the CDB data structure) to establish a connection. For at least this additional reason, independent claims 1 and 17 are not anticipated by Elzur and are in condition for allowance. Accordingly, dependant claims 2-16 and 18 are also not anticipated and are in condition for allowance.

Independent Claim 1

Claim 1 recites a system comprising an interface logic configured to pre-configure a topology of nodes. The Office Action cites Elzur Fig. 6, and [0041], line 5, and [0010], line 9 as teaching the claim element. (Office Action, Page 2). However, the cited text does not teach the claim limitation. Figure 6 of Elzur teaches a multi tier architecture data center that may handle different traffic types

over a single fabric. (Elzur, page 3, paragraph [0039]). However, Figure 6 fails to teach pre-configuring a topology of nodes. Note that pre-configuring a topology of nodes involves establishing membership in a group and that membership is later used to determine protocol actions like providing mapping data or making a connection using a preferred protocol. To the extent Elzur teaches any configuring, it only teaches writing values to SCSI interface registers. Elzur paragraph [0041], line 5 and paragraph [0010], line 9 also fail to teach pre-configuring a topology of nodes. The text mentions small computer system interface (SCSI). SCSI is a set of standards for physically connecting and transferring data between computers and peripheral devices. However, the cited text makes no mention of pre-configuring a topology of nodes.

Claim 1 also recites a mapping logic. The Office Action cites Elzur paragraph [0043], line 2 and paragraph [0008], lines 5-6 as teaching a mapping logic. However, the CDB taught in Elzur is not the mapping logic of claim 1. A CDB or command descriptor block is a block of bytes used in SCSI to send commands. Sending commands in SCSI does not anticipate a mapping logic. Therefore, Elzur does not anticipate claim 1, leaving it in condition for allowance.

Claim 2

This claim depends from claim 1, which has been shown not to be anticipated and thus this claim is similarly not anticipated. Furthermore, this claim recites the additional elements of acquiring a node identifier and topology configuration choice data. This data is acquired to facilitate the group membership decisions. Since Elzur is unconcerned with group membership, it follows that Elzur acquires and processes none of this data. The Office Action asserts that the CDB and context for a connection described in [0043] describe the additional elements. This is incorrect. Since Elzur is not concerned with pre-configuring a topology, it follows that Elzur does not teach acquiring identifiers and configuration choice data and then pre-configuring based on this data. In fact, the Elzur reference is directed toward a conventional system that has nothing to do with topology configuration

using "choice data" as recited in claim 2. Additionally, the claim recites providing "a topology data concerning the topology to a member of the topology." Neither [0041] or [0043] teach this element. For these additional reasons this claim is not anticipated and is in condition for allowance.

Claim 7

This claim depends from claim 2, which has been shown not to be anticipated and thus this claim is similarly not anticipated. Furthermore, this claim further characterizes the topology data. Since Elzur does not process topology data as claimed and described, it follows that Elzur does not further characterize this missing data. In particular, Elzur is completely silent concerning specifying a fallback network protocol and a fallback path. The fallback protocol and path are made available to non-members of the pre-configured topology. Since Elzur is not concerned with group membership, no fallback processing for non-members of a group is taught by Elzur. The rejection simply refers to Fig. 6, which shows neither of these elements. For this additional reason this claim is not anticipated and is in condition for allowance.

Claim 8

This claim depends from claim 1, which has been shown not to be anticipated and thus this claim is similarly not anticipated. Furthermore, this claim recites the additional element of the interface logic controlling resource control actions. To the extent that any of these actions are controlled by the system in Elzur, they appear to be controlled by a central processing unit on a mother board, not on an RNIC as claimed and described. (e.g., Elzur, page 3, paragraph [0038]) For this additional reason this claim is not anticipated and is in condition for allowance.

Claim 14

This claim depends from claim 2, which has been shown not to be anticipated and thus this claim is similarly not anticipated. Furthermore, this claim recites the additional element of having the connection management logic exert its control based on analyzing topology data in combination with other data (e.g., time of day, load,...). Elzur does not even analyze basic topology data because Elzur is not concerned with membership in a pre-configured topology. Therefore it follows that Elzur does not analyze topology data in light of additional factors (e.g., time of day) that may affect group performance. For this additional reason this claim is not anticipated and is in condition for allowance.

Claim 17

This claim recites the interface logic, mapping logic, and connection management logic of claim 1, which has been shown not to be anticipated, along with additional elements described in claims 2 and 3. Both claims 1 and 2 have been shown to be not anticipated and thus this claim is similarly not anticipated. Additionally, this claim recites the connection management logic exerting its control based on analysis of topology data including a node identifier and topology configuration choice. The reference does not teach this. For this additional reason this claim is not anticipated and is in condition for allowance.

Independent Claim 18

Claim 18 describes a method that includes acquiring a set of node identifiers, establishing a pre-configured topology, and providing membership data. The Office Action asserts that Figure 6 & 7, the CDB, and paragraphs [0043] and [0023] teach all these elements. While these figures and paragraphs clearly describe a network and a method for network interfacing, they are silent concerning establishing a pre-configured topology of nodes as claimed and described. None of the recited elements collect data associated

with group membership in a pre-configured topology of nodes, establish membership in the group, and then control actions (e.g., providing membership data) based on group membership. For at least these reasons claim 18 is not anticipated and is in condition for allowance.

Independent Claim 23

Claim 23 describes a method that includes acquiring a set of node identifiers, establishing a pre-configured topology, distributing membership data about the pre-configured topology to members of the pre-configured topology, selectively adding or deleting a node and then redistributing membership data, and selectively managing a resource and then once again redistributing membership data. Elzur merely describes a computer network and a method for interfacing in the network. None of the recited elements collect data associated with group membership in a pre-configured topology of nodes, establish membership in the group, and then control actions (e.g., providing membership data) based on group membership. Since Elzur is silent about a pre-configured topology, it follows that Elzur also does not describe (re)distributing topology data as nodes are added/deleted and/or as resources are managed. Claim 23 is therefore not anticipated for at least these reasons and are in condition for allowance. Accordingly, dependent claims 24-26 are also not anticipated for at least these reasons and are also in condition for allowance.

Claim 25

This claim depends from claim 23, which has been shown not to be anticipated and thus this claim is similarly not anticipated. Furthermore, this claim recites the additional element of establishing both a preferred and a fallback network protocol and path. To the extent that Elzur describes any protocol or path, it is limited to a single protocol and path per connection request. The fallback protocol and path are made available to non-members of the pre-configured

topology. Since Elzur is not concerned with group membership, no fallback processing for non-members of a group is taught by Elzur. Claim 25 is not anticipated for at least this additional reason, leaving claim 25 in condition for allowance.

Claim 26

This claim depends from claim 23, which has been shown to be not anticipated and thus this claim is similarly not anticipated. This claim recites the additional element of managing the networking resource. The management includes enabling an off-load capability, aging an off-loaded connection, converting an idle connection to a non-off-load mode, and converting a connection between an RDMA mode and a non-RDMA mode. These actions all facilitate preventing denial of service attacks. While Elzur describes a flow through NIC that may perform RDMA or off-load capability, these states appear to be static in the reference and not individually dynamically manageable as claimed and described in claim 26. Elzur therefore teaches none of the resource management that could prevent the denial of service attacks as described in at least paragraphs 18 and 19 of the application as originally filed. For this additional reason this claim is not anticipated and is in condition for allowance.

Independent Claim 27

Claim 27 recites a method that includes selectively not establishing a connection between nodes if a node is determined not to be a member of a pre-configured topology. Elzur does not concern a pre-configured topology as claimed and described. Therefore, it clearly does not decide on connection establishment based on topology membership. The reference simply describes a converged network controller (CNC) that presumably handles TCP/IP requests in a conventional manner that does not include pre-configured topology membership processing. For at least this reason claim 27 is not anticipated and is in condition

for allowance. Accordingly, dependent claims 28-29 are similarly not anticipated and are in condition for allowance.

Independent Claim 30

Claim 30 stands rejected using the same rationale as claim 27 and is, therefore, not anticipated for the same reasons provided above. Additionally, claim 30 is a Beauregard claim while Elzur does not appear to describe a computer-readable medium on which any set of instructions that when executed can cause a processor to perform any method, let alone a topology membership-based connection management method as claimed and described. For this additional reason claim 30 is not anticipated and is in condition for allowance.

III. Whether claims 4-6, 10, 19-22, 31-36, and 39 are unpatentable under 35 U.S.C. 103(a) as being obvious over Elzur in view of Delany.

**Claims 4-6, 10, 19-22, 31-36 and 39** were rejected under 35 U.S.C. §103(a) as being unpatentable over Elzur in view of Delany. To establish a prima facie case of 35 U.S.C. §103 obviousness the prior art reference (or references when combined) must teach or suggest all the claim limitations. MPEP 2143.03 Here, the criteria described in MPEP 2143.03 is not satisfied since the combination of references does not teach or suggest all the claim limitations. None of the references, alone and/or in combination, teach anything to do with pre-configuring a topology of nodes to establish membership in a group. Additionally, none of the references, alone and/or in combination teach selectively making connection requests on either a preferred or fallback protocol based on membership in the pre-configured topology. Thus, none of the claims are obvious for at least this reason.

Claims 4-6

Claims 4-6 depend from claim 1, which has been shown to be not anticipated by Elzur. Therefore, claims 4-6 cannot be obvious in light of Elzur and Delany. Claim 4 concerns a system that performs a connection management based on membership in a pre-configured topology. Neither Elzur nor Delany concern establishing membership in a pre-configured topology. Claim 4 further characterizes an identifier that can be processed to determine topology membership. Since neither Elzur nor Delany process topology membership, it follows that neither further characterizes an identifier processed to determine membership.

The Office Action asserts that Delany teaches that a node identifier may be a value stored on a Universal Serial Bus (USB) token. The Office Action cites col. 7, line 47 to support this assertion. The passage reads "the system 200 may also communicate with local occasionally-connected devices ... which may include an RS-232 serial port, a USB interface, or the like." While the words USB appear in

the passage, both the passage and the reference are entirely void of any description of a node identifier stored on a USB token. Making a final rejection of a claim by twice rejecting the claim based on the mere appearance of a word in a sentence, especially when the inadequacy has been specifically called out in response to an Office Action, does not provide Applicant with adequate opportunity to advance prosecution on the merits. Additionally, the simple appearance of words does not "teach" an element. For this additional reason this claim should not be rejected and is in condition for allowance.

Claim 6

This claim depends from claim 2, which has been shown to be not anticipated, and thus this claim cannot be obvious over the same reference in light of Delany. Furthermore, this claim recites the additional elements of pre-configuring including establishing both preferred and fallback protocols and paths. The fallback protocol and path are made available to non-members of the pre-configured topology. Since Elzur is not concerned with group membership, no fallback processing for non-members of a group is taught by Elzur. The Office Action asserts that Delany teaches establishing both preferred and fallback protocols and paths. This is unsupported and incorrect. Delany describes how email clients can be connected over a network to at least one message transfer agent (MTA) (col. 1, lines 28-31) and how a mailing list manager should have reliable fallback MTAs available (col. 47, lines 12-30). A description of configuring an email network with a fallback MTA in no way touches on topology pre-configuring that includes establishing both preferred and fallback protocols and paths. For this additional reason claim 6 is not obvious and is in condition for allowance.

Claim 10

This claim depends from claim 1, which has been shown to be not anticipated and therefore this claim cannot be obvious over the same reference in

light of Delany. Furthermore, claim 10 recites the mapping logic establishing a fallback connection according to a second protocol that is different from a first requested protocol. The fallback protocol and path are made available to non-members of the pre-configured topology. Since Elzur is not concerned with group membership, no fallback processing for non-members of a group is taught by Elzur. While Delany describes having a fallback MTA available for email, this does not teach switching protocols to establish a connection on a second protocol after a first connection request using a first protocol has been denied. For this additional reason this claim is not obvious and is in condition for allowance.

Claim 19

This claim depends from claim 18, which has been shown to be not anticipated and thus this claim cannot be obvious over the same reference and Delany. Furthermore, this claim recites receiving node identifiers from a human through a GUI, from a scripting system, or from a policy-based system. The Office Action admits that Elzur does not teach receiving node identifiers from any of these sources. The Office Action then relies on Delany to cure the deficiency in Elzur. The Office Action asserts that Delany teaches a GUI and therefore it would be obvious to combine the references to facilitate receiving user commands and data. While interesting, this is irrelevant because the claim concerns receiving node identifiers associated with pre-configured topology membership not user commands and data as taught in the GUI reference. For this additional reason this claim is not obvious and is in condition for allowance.

Claim 20

This claim depends from claim 18, which has been shown to be not anticipated and thus this claim cannot be obvious over the same reference and Delany. Furthermore, this claim concerns performing a connection management based on membership in a pre-configured topology. Neither Elzur nor Delany concern establishing membership in a pre-configured topology. Claim 20 further

characterizes an identifier that can be processed in determining topology membership. Since neither Elzur nor Delany process topology membership, it follows that neither further characterizes an identifier processed to determine membership.

The Final Office Action asserts that Delany teaches that a node identifier may be a value stored on a Universal Serial Bus (USB) token. The Office Action cites col. 7, line 47. The passage reads "the system 200 may also communicate with local occasionally-connected devices ... which may include an RS-232 serial port, a USB interface, or the like." As described above, the mere appearance of the word "USB" in the passage does not create a prima facie case of obviousness. The reference is entirely void of any description of a node identifier stored on a USB token. For this additional reason this claim is not obvious and is in condition for allowance.

Claim 21

This claim depends from claim 18, which has been shown to be not anticipated and thus this claim cannot be obvious over the same reference and Delany. Furthermore, this claim recites how pre-configuring includes establishing both preferred and fallback protocols and paths. The Office Action asserts that Delany teaches establishing both preferred and fallback protocols and paths. This is wrong. Delany describes how email clients can be connected over a network to at least one message transfer agent (MTA) (col. 1, lines 28-31) and how a mailing list manager should have reliable fallback MTAs available (col. 47, lines 12-30). Configuring an email network with a fallback MTA in no way touches on topology pre-configuring that includes establishing both preferred and fallback protocols and paths. For this additional reason claim 21 is not obvious and is in condition for allowance.

Claim 22

This claim was rejected by reference to claims 18 and 21 and is, therefore, not obvious for the same reasons provided above.

Independent Claim 31

This claim recites a method that includes several actions. The actions include receiving a mapping request, selectively providing mapping data to the requester upon determining that it is a member of a pre-configured topology, receiving a connection request from the request and selectively establishing a connection using a first protocol upon determining that the requester is a member of the pre-configured topology. Note that there are multiple steps. The method listens at a well-known port for a mapping request and only hands over the mapping data after determining that the requester is entitled to the mapping data. Then, the method only provides a connection when an additional request has been validated. This differs from the conventional approaches hinted at in Elzur.

The method also includes receiving a fallback connection request to establish a connection using a second (e.g., fallback) protocol, where the connection will not provide access to a resource that would be available over the first (e.g., preferred) protocol over the first (e.g., preferred) connection.

The Office Action simply asserts that the rejections of claims 1 and 10 provide the rationale for this rejection. This "rejection by reference" yields an incomplete examination and does not provide Applicant with an adequate opportunity to reply since not every element of claim 31 has been considered. Specifically, claim 31 describes the fallback connection not providing access to a resource that is reachable through the first protocol. In addition to the procedural problem with the rejection, there is a substantive problem. As has been described above, neither Elzur nor Delany teach establishing a connection using a fallback (different) protocol that does not give access to a resource that is available through a first protocol. For this additional reason this claim is not obvious and is in condition for allowance.

Claim 32

This claim depends from claim 31, which has been shown to be not obvious and thus this claim is similarly not obvious. Furthermore, this claim recites establishing a connection based on the availability of a resource. Neither Elzur nor Delany describe this element and no citation is provided to any portion of either reference to support the rejection. For this additional reason this claim is not obvious and is in condition for allowance.

Independent Claim 36

This claim is a Beauregard claim corresponding to independent claim 31. Claim 31 has been shown to be not obvious and therefore this claim is similarly not obvious. Furthermore, this is a computer-readable medium claim and neither Elzur nor Delany appear to describe storing computer executable instructions on a computer-readable medium. Indeed, it would appear to be physically impossible to craft a Beauregard claim relying on the specific registers and data structures relied on by the Final Office Action. For this additional reason this claim is not obvious and is in condition for allowance.

## Conclusion

For the reasons set forth above, a prima facie anticipation or obviousness rejection has not been established for any claim. All rejections have been shown to be improper. Appellant respectfully believes that all pending claims **1-39** patentably and unobviously distinguish over the references of record and that the rejections should be reversed. Appellant respectfully requests that the Board of Appeals overturn the Examiner's rejections and allow all pending claims. An early allowance of all claims is earnestly solicited.

Respectfully submitted,

9-29-08

Date

Peter Kraguljac (Reg. No. 38,520)
John T. Kalnay (Reg. No. 46,816)

(216) 308-3245
Kraguljac & Kalnay, LLC
Summit One, Suite 510
4700 Rockside Road.
Independence, OH 44131

## 8. Claims Appendix

1. A system, comprising:

an interface logic configured to pre-configure a topology of nodes to communicate via a preferred networking protocol;

a mapping logic operably connected to the interface logic, the mapping logic being configured to produce a mapping between a resource located on a first node and a port located on the first node, to selectively provide to a second node a mapping data that describes the mapping, and to selectively establish a connection that facilitates the second node accessing the resource through the port using the preferred networking protocol; and

a connection management logic operably connected to the mapping logic and the interface logic, the connection management logic being configured to control whether the mapping logic will provide the mapping data and establish the connection.

2. The system of claim 1, where to pre-configure the topology of nodes the interface logic acquires a node identifier that facilitates recording whether a node is a member of a pre-configured topology, acquires a topology configuration choice data concerning how the pre-configured topology is to be configured, pre-configures the topology based, at least in part, on the node

identifier and the topology configuration choice data, and provides a topology data concerning the topology to a member of the topology.

3. The system of claim 2, where the connection management logic exerts its control based, at least in part, on the topology data and a node identifier received from the second node.

4. The system of claim 2, where a node identifier comprises one or more of, an Internet Protocol (IP) address, a value stored in one or more of a network interface card (NIC) hardware, firmware, and software, a value stored in one or more of a remote direct memory access (RDMA) NIC (RNIC) hardware, firmware, and software, a password, and a value stored on a universal serial bus (USB) token.

5. The system of claim 1, where the configuration choice data is received from one or more of, a human user via a graphical user interface (GUI), a scripting-based system, and a policy-based system.

6. The system of claim 2, where to pre-configure the topology of nodes, the interface logic determines which nodes are members of the topology, establishes a preferred computer networking protocol to be employed by members of the topology, establishes a preferred path to be employed for data

communications between members of the topology, establishes a fallback networking protocol to be employed by members of the topology, and establishes a fallback path to be employed for communications between members of the topology.

7. The system of claim 2, where the topology data describes one or more of, which nodes are members of the topology, a preferred computer networking protocol to be employed between members of the topology, a preferred path to be employed for communications between members of the topology, a fallback networking protocol to be employed between members of the topology, and a fallback path to be employed for communications between members of the topology.

8. The system of claim 1, where the interface logic is further configured to control one or more resource control actions including, enabling a protocol off-load capability, aging off-loaded connections, converting idle connections to a non-off-load mode, and converting connections between an RDMA and a non-RDMA mode.

9. The system of claim 1, where the mapping logic comprises a port mapper configured to listen on a well-known port for one or more of, a request for mapping data, and a connection request.

10. The system of claim 1, the mapping logic being further configured to facilitate establishing a fallback connection between the first node and the second node according to a second networking protocol, the second networking protocol being different from the first networking protocol, where the second node may request the fallback connection after the mapping logic has been controlled to not provide the mapping data to the second node or the mapping logic has been controlled to prevent the establishment of a connection between the first node and the second node using the first networking protocol.

11. The system of claim 10, the connection management logic being configured to block access to a first resource on the first node via the preferred networking protocol and to permit access to a second resource on the first node via a fallback networking protocol.

12. The system of claim 1, where the resource supports one or more of, remote direct memory access (RDMA) between the first node and the second node, and protocol off-loading at the first node.

13. The system of claim 1, where one or more of, the interface logic, the mapping logic, and the connection management logic are located on one or

more of, a network interface card (NIC), and a remote direct memory access (RDMA) NIC (RNIC).

14. The system of claim 2, where the connection management logic exerts its control based on analyzing the topology data and one or more of, time of day, network traffic, load, and resource availability.

15. The system of claim 1, where the connection management logic operates at a session layer associated with the first networking protocol.

16. The system of claim 15, where the first networking protocol includes a Transmission Control Protocol (TCP) transport layer and an Internet Protocol (IP) network layer.

17. A computer configured with a pre-configured topology connection management system, the system comprising:

an interface logic configured to pre-configure a topology of nodes to communicate via a preferred networking protocol or a fallback networking protocol, where to pre-configure the topology of nodes the interface logic acquires a node identifier that facilitates recording whether a node is a member of a pre-configured topology, acquires a topology configuration choice data

concerning how the pre-configured topology is to be configured, pre-configures the topology based, at least in part, on the node identifier and the topology configuration choice data, and provides a topology data concerning the topology to a member of the topology;

a mapping logic operably connected to the interface logic, the mapping logic being configured to produce a mapping between a resource located on a first node and a port located on the first node, to selectively provide to a second node a mapping data that describes the mapping between the resource and the port, and to selectively establish a connection between the first node and the second node, where the connection facilitates the second node accessing the resource through the port using the preferred networking protocol; and

a connection management logic operably connected to the mapping logic and the interface logic, the connection management logic being configured to control whether the mapping logic will provide the mapping data to the second node, and whether the mapping logic will establish the connection, where the connection management logic exerts its control based, at least in part, on the topology data and a node identifier received from the second node.

18. A method, comprising:

acquiring a set of node identifiers associated with nodes to be considered for inclusion in a pre-configured topology of nodes that can communicate within the topology using a preferred computer networking protocol;

establishing the pre-configured topology of nodes; and

making available a membership data concerning the pre-configured topology of nodes.

19. The method of claim 18, where the set of node identifiers are acquired from one or more of, a human user through a graphical user interface (GUI), a scripting-based system, and a policy-based system.

20. The method of claim 19, where a node identifier comprises one or more of, an Internet Protocol (IP) address, a value stored in a network interface card (NIC) hardware, a value stored in a NIC firmware, a value stored in a NIC software, a value stored in a remote direct memory access (RDMA) NIC (RNIC) hardware, a value stored in an RNIC firmware, a value stored in an RNIC software, a password, and a value stored on a USB (Universal Serial Bus) token.

21. The method of claim 18, where establishing the pre-configured topology of nodes includes:

determining node membership in the pre-configured topology;

establishing a preferred computer networking protocol to be employed by members of the topology;

establishing a preferred path to be employed for communications between members of the topology;

establishing a fallback computer networking protocol to be employed by members of the topology;

establishing a fallback path to be employed for communications between members of the topology; and

recording the topology membership, preferred computer networking protocol, preferred path, fallback computer networking protocol, and fallback path in the membership data.

22. A computer-readable medium storing processor executable instructions operable to perform a method, the method comprising:

acquiring a set of node identifiers associated with nodes to be considered for inclusion in a pre-configured topology of nodes that can communicate within the topology using a preferred computer networking protocol or a fallback computer networking protocol;

establishing the pre-configured topology of nodes, where establishing the pre-configured topology of nodes includes determining node membership in the pre-configured topology, establishing a preferred computer networking protocol to be employed by members of the topology, establishing a preferred path to be employed for communications between members of the topology, establishing a fallback computer networking protocol to be employed by members of the topology, establishing a fallback path to be employed for communications between members of the topology, and recording the topology membership, preferred computer networking protocol, preferred path, fallback computer networking protocol, and fallback path in the membership data; and

making available a membership data concerning the pre-configured topology of nodes.

23. A method, comprising:

acquiring a set of node identifiers associated with nodes to be considered for inclusion in a pre-configured topology of nodes that can communicate within the topology using a preferred computer networking protocol;

establishing the pre-configured topology of nodes;

distributing a membership data concerning the pre-configured topology of nodes to nodes that are in the pre-configured topology of nodes;

selectively adding or deleting a node from the pre-configured topology of nodes and, in response to selectively adding or deleting the node, redistributing the membership data; and

selectively managing a computer networking resource, and in response to selectively managing the computer networking resource, redistributing the membership data.

24. The method of claim 23, where a node identifier comprises one or more of, an Internet Protocol (IP) address, a value stored in a network interface card (NIC) hardware, a value stored in a NIC firmware, a value stored in a NIC software, a value stored in a remote direct memory access (RDMA) NIC (RNIC) hardware, a value stored in an RNIC firmware, a value stored in an RNIC software, a password, and a value stored on a USB token.

25. The method of claim 23, where establishing the pre-configured topology of nodes includes:

determining node membership in the pre-configured topology;

establishing a preferred computer networking protocol to be employed by members of the topology;

establishing a preferred path to be employed for communications between members of the topology;

establishing a fallback computer networking protocol to be employed by members of the topology;

establishing a fallback path to be employed for communications between members of the topology; and

recording the group membership, preferred computer networking protocol, preferred path, fallback computer networking protocol, and fallback path in the membership data.

26. The method of claim 23, where selectively managing a computer networking resource includes one or more of, enabling a protocol off-load capability, aging an off-loaded connection, converting an idle connection to a non-off-load mode, and converting a connection between an RDMA mode and a non-RDMA mode.

27. A method, comprising:

in a first node, receiving from a second node, via an open computer networking protocol, a request to establish a connection between the first node and the second node via the open computer networking protocol, where the connection facilitates the second node accessing a resource associated with the first node;

determining whether the second node is a member of a pre-configured topology that includes the first node by examining a node identifier associated with the second node; and

selectively not establishing the connection between the first node and the second node via the open computer networking protocol if it is determined that the second node is not a member of the pre-configured topology that includes the first node.

28. The method of claim 27, where the method is performed by a session layer logic associated with the open computer networking protocol.

29. The method of claim 27, where the open computer networking protocol includes a Transmission Control Protocol (TCP) transport layer and an Internet Protocol (IP) network layer.

30. A computer-readable medium storing processor executable instructions operable to perform a method, the method comprising:

in a session layer logic in a first node, receiving from a second node, via an open computer networking protocol that includes a Transmission Control Protocol (TCP) transport layer and an Internet Protocol (IP) network layer, a request to establish a connection between the first node and the second node via the

open computer networking protocol, where the connection facilitates the second node accessing a resource associated with the first node;

determining whether the second node is a member of a pre-configured topology that includes the first node; and

selectively not establishing the connection between the first node and the second node via the open computer networking protocol if it is determined that the second node is not a member of the pre-configured topology that includes the first node.

31. A method, comprising:

in a first node, receiving from a second node a mapping request for a mapping data that describes a relationship between a resource on the first node and a port on the first node;

selectively providing the mapping data to the second node based on determining that the second node is a member of a pre-configured topology that includes the first node by examining a node identifier associated with the second node;

receiving from the second node a connection request to establish a connection between the first node and the second node via a first networking protocol, where the connection facilitates accessing the resource;

selectively establishing the connection based on determining that the second node is a member of a pre-configured topology that includes the first node by examining a node identifier associated with the second node; and

via a second networking protocol, receiving from the second node a fallback connection request to establish a fallback connection between the first node and the second node, where the fallback connection request requests that the fallback connection be established via the second networking protocol, where the fallback connection granted in response to the third request will not provide access to the resource via the first networking protocol.

32. The method of claim 31, where selectively establishing the connection is based additionally on an availability of the resource.

33. The method of claim 31, where the resource is located on one or more of, a network interface card (NIC), and a remote direct memory access (RDMA) NIC (RNIC) associated with the first node.

34. The method of claim 31, where the resource supports one or more of remote direct memory access (RDMA) between the first node and the second node, and protocol off-loading at the first node.

35. The method of claim 31, where the first networking protocol includes a Transmission Control Protocol (TCP) transport layer and an Internet Protocol (IP) network layer.

36. A computer-readable medium storing processor executable instructions operable to perform a method, the method comprising:

in a first node, receiving from a second node a mapping request for a mapping data that describes a relationship between a resource on the first node and a port on the first node;

selectively providing the mapping data to the second node based on determining, by examining a node identifier associated with the second node, that the second node is a member of a pre-configured topology that includes the first node;

receiving from the second node a connection request to establish a connection between the first node and the second node via a first networking protocol, where the connection facilitates accessing the resource;

selectively establishing the connection based on determining that the second node is a member of a pre-configured topology that includes the first node by examining a node identifier associated with the second node; and

via a second networking protocol, receiving from the second node a fallback connection request to establish a fallback connection between the first node and the second node, where the fallback connection request requests that the fallback connection be established via the second networking protocol, where the fallback connection granted in response to the third request will not provide access to the resource via the first networking protocol.

37. A system, comprising:

means for determining whether a client node is a member of a pre-configured topology to which a server node belongs;

means for rejecting a request that will lead to the undesired consumption of a server resource if the requesting client node is not a member of the pre-configured topology to which the server node belongs; and

means for establishing a connection between the client node and the server node using a networking protocol preferred by members of the pre-configured topology.

38. A set of application programming interfaces embodied on a computer-readable medium for execution by a computer component in conjunction with pre-configured topology connection management, comprising:

a first interface for communicating a group data configured to facilitate determining whether a client node is a member of a pre-configured topology to which a server node belongs; and

a second interface for communicating a resource management data that facilitates determining whether a client node will be granted a connection to a resource located on a topology member node.

39. The system of claim 2, where the topology configuration choice data is received from one or more of, a user, a script, a policy, and a program.

## 9. Evidence Appendix

None. There is no extrinsic evidence.

## 10. Related Proceedings Appendix

None. There are no related proceedings.